10/537650
Rec'd PCT/PTO 18 APR 2006

TRANSLATION (SDS-0119):

WO 2004/053756 A2

PCT/IB2003/005680

FINANCIAL PRODUCT PRICING SYSTEM

The invention concerns a financial product pricing system.

Financial products, and in particular derivative products, are subject to very strict regulation that requires financial establishments and commercial enterprises using them to price them as a function of a certain number of hypotheses, involving variables arising from the characteristics of the products in question and variables arising from the financial markets to which these products belong. These prices must be calculated for the purposes of valuation and also so that they can be incorporated into the required Financial Risk reports, in particular the VaR (Value at Risk). These "pricing" calculations must be updated very frequently -- usually daily -- which involves very cumbersome data processing.

In addition, many new financial products appear regularly. The analysis of each new product in order to calculate the price, the VaR, etc., is very cumbersome work, and most often requires developing a specific pricing model. This model is then integrated into the information processing systems used for the examination, processing, valuation, and calculation of risks. These systems are often different.

The invention aims to provide a system that makes it possible to simplify considerably the operations involved in pricing a financial product.

For this purpose, the object of the invention is a financial product pricing system consisting of interface means, data storage means, calculation means, and data processing means, a system that is remarkable in that:

(a) the interface means consist of means for inputting data that identify and describe the product into the system, whereby these data consist of:

-- (a1) contextual data of the product, consisting of at least one valuation currency and at least one underlying instrument,

-- (a2) characteristic data of the product, consisting of a set of events and flows associated with the product;

(b) the data processing means consist of means for generating a planned schedule from the identification and description data, in which an event and/or flow relating to the product is associated with each date;

(c) the data processing means also include means for interpreting the schedule, in order to generate:

-- (c1) a table of variables for the product on the basis of the events and/or flows,

-- (c2) for each date of the planned schedule, a function for calculating the product price as a function of at least one of the product variables;

(d) the interface means consist of means for inputting a list of market variables associated with the product and generated by a market analysis; and

(e) the calculation means consist of means for calculating, for each of the market scenarios/states and for each of the dates, the product variable values according to the market variables, and means for calculating the product price as a function of the calculated product variable values.

According to one embodiment, the data processing means consist of means for generating a compact script containing all the data needed for product pricing.

According to another embodiment, the means for inputting data identifying and

describing the product consist of means for inputting these data in compact script form.

According to another embodiment, the means for inputting data identifying and describing the product consist of acquisition windows, into which the contextual data and characteristic data can be entered separately.

According to another characteristic of the invention, the data processing means also include means for checking the interpretation of the schedule.

According to yet another characteristic of the invention, the calculation means consist of:

-- (e1) means for calculating, for each of the market scenarios/states and for each of the dates, the value of each of the market variables,

-- (e2) means for calculating, for each of the market scenarios/states and for each of the dates, the product variable values as a function of the market variable values,

-- (e3) means for calculating the price as a function of the product variable values in all the market scenarios/states.

Other characteristics and advantages of the invention will become evident from the following description of an embodiment of the invention, illustrated by the attached drawings, in which:

-- Figure 1 is a functional block diagram illustrating the basic means used by the system according to the invention to describe a financial product and to price it;

-- Figure 2 is a functional block diagram illustrating the means and data needed to price a financial product by the system according to the invention;

-- Figure 3 is a functional block diagram illustrating the process for pricing a financial product using the system according to the invention;

-- Figures 4 and 5 are tables illustrating the storage means needed for the pricing process

of Figure 3;

-- Figures 6 to 8 illustrate the computer acquisition windows for the information needed to describe a financial product according to the invention.

Referring to Figure 1, pricing a financial product using the system according to the invention involves the following data and means:

-- a calculation tool called a pricer 1;

-- product variables 2 that are a function of the nature of the financial product in question (for example, swap, call, etc.);

-- market hypotheses 3 known to financial specialists (for example, normal, log-normal, mean reverting, multifactor, etc.) that depend, among other things, on the underlying instrument 4 (for example, share, bond, exchange rate, credit, etc.) attached to the product in question; these market hypotheses make it possible to determine the various market scenarios or states and the way in which these scenarios will be generated and modeled, whereby each scenario corresponds to a set of possible market variable values, a set that is used to price the financial product;

-- calculation means consisting of numerical resolution methods known to financial specialists (for example, trees 6, integration 7, partial differential equations or PDE 8, the Monte Carlo method 9, etc.), which make it possible to price the product on the basis of the product variables and market hypotheses.

According to Figure 2, the system making it possible for a user to price a financial product first involves a description of the product. This first of all requires acquisition of the following, thanks to the system interface means:

-- contextual data of the product, that is, at least one valuation currency (making it possible to define all the conventions of the market or the place in question) and one or more

underlying instruments (for example, rate curve, “share”, exchange rate, etc.), that is, the market quantities used to define the product; these contextual data indicate what market variables will be involved in the product pricing and used for selecting market hypotheses;

-- characteristic data of the financial product in question. These data consist of a schedule of events and flows associated with the product.

For this purpose, the user may either:

-- input a compact script 10 of the type shown in Figure 8, which contains all the information (contextual data and characteristic data of the product) and will then be separated into different objects, each containing these two types of data, or

-- input directly into separate windows the product contextual data on the one hand and characteristic data on the other, as illustrated by the acquisition screen in Figure 6.

On the basis of these data describing and identifying the product, the data processing means of the system are used to generate a planned schedule at 13, that is, an exhaustive series of dates D1, D2 .... Dn, with an event and/or a flow relating to the product associated with each one (Table T1).

A table of objects T2 and a table of product variables T3 are constructed at 14 and 15, respectively, by means of a "parser".

Table T2 is an intermediate table that makes it possible to define and construct, for each date in question D1, D2 .... Dn, one or more evaluation trees representing a tree form of the function $Y_1 = f(t, x_1 .... x_n, Y_{t-1})$, in which:

-- Y_1 represents the current value of the product variable or variables;

-- t represents time;

-- $x_1 .... x_n$ represents the value of the market variables that will be identified at 17.

Table T3 is a list of product variables and the values associated with each of these variables, values that are updated by the evolution tree or trees of Table T2.

At 16 the system uses verification means to make sure of the consistency of the data of Table T2.

At 17, a market analysis is done in order to construct an objects table, "world" (T4). This table consists of a list of additional market information that is needed at each of the dates D1 to Dn to price or evaluate the product.

These "world" objects are defined on the basis of a number of basic elements, four of which are especially important:

-- the spot (shares, raw materials, energy, indexes, etc.);

-- the exchange rate;

-- the interest rate, preferably represented by discount factors. This can also be zero-coupon rates, the Libor rate, a swap, etc.;

-- information on counterparty default; for example, has a bond defaulted? This may also involve a rate "spread" representing the credit risk.

Since the market variables at this stage are identified and Table T4 is constructed, construction of the evaluation trees of Table T2 is therefore achieved.

On the basis of the data obtained as described with reference to Figure 2, the "pricer" 1 proceeds to calculate the price by applying one of the numerical resolution methods 5.

These financial numerical resolution methods (for example, trees 6, integration 7, partial differential equations or PDE 8, the Monte Carlo method 9, etc.), which are standard and well known to finance specialists, achieve the following:

-- simulate or explore possible values of market variables;

-- calculate the desired or future value of product variables.

The functional block diagram in Figure 3 illustrates the numerical resolution of the product pricing problem. At 20, depending on the numerical resolution method 5 in question, acquisition of the contextual data used in the method (which have been obtained as described with reference to Figure 2) and of the number of product variables is carried out.

At 21, numerical resolution means generate the values of the market variables at each date D1, D2 .... Dn of the schedule according to the market hypotheses in question, as well as at each scenario established as a function of these hypotheses. As shown in Figure 4, a table of market variable values Tvvm corresponds to each date and market scenario/state.

At 22, the numerical resolution means calculate the product variable values for each date and market scenario/state in question. As shown in Figure 5, a table of product variable values Tvp corresponds to each date and market scenario/state in question.

At 23, the numerical resolution means finally produces a product price as a function of the set of calculated product variable values.

An example of implementing the mechanisms making it possible to identify and describe a financial product by its contextual data and its characteristic data as indicated at 10 and 12, in order to construct Table T1 in Figure 2, will now be described in greater detail with reference to Figures 6 to 8, which represent acquisition windows of the product to be described.

These mechanisms make it possible to describe any structured financial product, independently of its underlying instrument or the structure or characteristics of its flows. These mechanisms have:

-- a syntax defining the type of phrase structure accepted and compressed;

-- a dictionary of predefined words that are "compressed";

-- the capacity to accept new words if they are suitably defined and introduced.

In the example that follows, a convertible bond will be defined. First, the market variables in question, that is, the currency and its "rate" curve and the pertinent "transferable security", in this case the DAX, are introduced into the windows 30 and 31 of Figure 6.

It is necessary to describe the sum that the bond will pay at maturity if it is not converted. For this purpose, the term "Redempt" is introduced at 32, to designate the amount paid back at maturity. A numerical value is assigned to it at 33, in this case 100.

Likewise, "Coupon" and "ConvPrice" are introduced at 34 and 35, and their respective numerical values at 36 and 37.

In order to suitably define the product, the idea of "conversion ratio" must be introduced. This is done at 38 by indicating that "Conv_Ratio(x)" is equal to "100*(x)/ConvPrice" (window 39).

Then we must describe the product, that is, the flows that it will generate and the conditions of this generation if necessary.

In the hypothesis in question, the convertible bond pays a coupon (window (43) from any point of departure (window 40), at an annual frequency (window 42) for five years (window 41).

At the end of five years (window 44), the bond pays its redemption value (window 45).

The bond's conversion characteristic is expressed by the fact that at any time ("Od", window 46) during the five years (window 47), the value of the product is the maximum of the product value and its conversion ratio (window 48):

Convert = max (convert, Conv_Ratio(dax)).

These mechanisms also make it possible to very simply describe a convertible bond with particularly unusual characteristics.

Thanks to the interface means, shown in Figure 7, the system according to the invention generates and can then display the product flows based on the data introduced in the form of a preestablished format. This allows the user to make sure that the discounted flows are properly represented and captured by the system.

Finally, the data processing means make it possible for the system to generate a script (Figure 8), that is, a code precisely describing the product characteristics and containing all the information necessary for pricing the product. The script, show in Figure 8, can be exchanged among all the intervening parties to describe and price the product.

As a variant, all the information, that is, the contextual data and the characteristic data of the product in question, can be acquired by a user in the form of a compact script of the type shown in Figure 8, which can then be broken down into different objects, each containing the product's contextual data and characteristic data.

Describing the product using windows, leading to the generation of a script that can be exchanged among and interpreted by different computers as described above, or directly in the form of a script, requires the user to use predefined words that are part of a dictionary that can be expanded, if necessary, and a syntax defining the type of phrase structure accepted and compressed.

The rules of syntax and the predefined vocabulary are given in an appendix, and an example of how to use this vocabulary and these rules of syntax is presented below. Thus the definition of the product value given in window 48:

Convert = max (convert, Conv_Ratio(dax))

is an assignment of variable using a function whose name is predefined "max", which is applied to two parameters:

-- the first parameter "convert" is a variable;

-- the second parameter "Conv_Ratio(dax)" makes use of a definition (of the macro function type) given in windows 38 and 39, which means that the expression "Conv_Ratio(dax)" should be replaced by the expression "100*dax/ConvPrice", in which "dax" is a variable and "ConvPrice" is an expression that should in turn be replaced by the numerical value "6500", because of the definition given it in windows 36 and 37.

The definition of window 48 uses a reserved vocabulary word "max", defined as being a function name. The set of reserved function names is defined in the grammar as follows:

```
FUNCTION_NAME =     "CONT"
                    | "MAX"
                    | "MIN"
                    | "MAXINDEX"
                    | "MININDEX"
                    | "SQRT"
                    | "LOG"
                    | "EXP"
                    | "ABS"
                    | "INT"
                    | "TERM"
                    | "SPOT"
                    | "FX"
                    | "DF"
                    | "LIBOR"
```

| “SWAPRATE”

| “LEVEL”

| “ACCRUEDCOUPON”

| “ACCRUALFACTOR”

| “BLACKSCHOLES”

| (“FIX_DATE” “FIXDATE”)

| (“START_PERIOD” “STARTPERIOD”)

| (“END_PERIOD” “ENDPERIOD”)

| (“PAY_DATE” “PAYDATE”)

| “CGV”

| “START”

| “END”

| “NOW”

The definition of window 48 also uses the following rules of syntax:

-- definition of an assignment:

assignment = ID “=” expression;

-- definition of a function:

function = (FUNCTION_NAME | auxTab_or_array) “(” exprList “)”;

-- definition of a list of expressions:

exprList = expression {“,” expression};

-- definition of an expression:

expression = sum;

sum = product { (“+” | “–”) product};

product = power { ("*" | "/") power};

power = sign { "^" sign};

sign = { "+" | "–" } atom;

atom= function | ID numconst "(" expression ")".

It is evident that the definition of an expression is given by rules successively dependent on each other and using the definition of a function for the last one. It is therefore possible in any expression to make use of a function that itself uses expressions, which also use functions, and so on recursively.